iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.71701

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Lidar SLAM for Mobile Robot in an Indoor Environment

Kambire Sie Stephane[1], Zhen Xu[2], Jin Ding[3], Yongqiang Xu[4]
[1, 2, 3]School of Automation and Electrical Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China
[4]Zhejiang Jinbang Sports Equipment Co., Ltd, Lishui 321400, China

Abstract: Two-dimensional (2D) simultaneous localization and mapping (SLAM) is a crucial technology for autonomous indoor robots. The robot can navigate and execute designated tasks using a map generated by SLAM. An indoor SLAM system utilizing LIDAR data that effectively tracks moving obstacles with notable accuracy and reliability. This paper presents a LIDAR-based SLAM methodology tailored for dynamic environments, specifically within indoor settings where changes occur due to mobile objects. The integration of LIDAR data with sophisticated algorithms facilitates high-precision localization and ensures the map remains current, even in intricate and confined spaces. The differentiation between static and dynamic feature extraction, along with adaptive filtering techniques, enhances localization accuracy and overall performance. Our experimental results are promising, showcasing consistent and safe navigation in dynamic indoor environments.
Keywords: Mobile robot; 2D Lidar; SLAM; Mapping

I. INTRODUCTION

Indoor robots have been widely utilized in contemporary manufacturing operations and domestic chores, encompassing automated guided vehicles (AGVs), logistics distribution, and household cleaning. Autonomous navigation is a fundamental technology for indoor robots [1,2]. SLAM generates an environmental map beforehand to facilitate robot navigation in various scenes [3,4]. In an unfamiliar location, a mobile robot can ascertain its position and orientation by gathering environmental data through sensors, concurrently creating a continuously updated map based on its pose. SLAM can be classified into vision-based and laser-based categories depending on the sensor type [5,6,7]. Currently, visual SLAM predominantly use RGBD cameras, monocular cameras, and binocular cameras. Prominent open-source visual SLAM systems encompass MonoSLAM [8], ORB-SLAM2 [9], and VINS-SLAM [10]. Visual SLAM delivers abundant texture data and diverse feature information; nevertheless, it is significantly influenced by ambient light and is ineffective in dark or non-textured environments. Due to its extensive field of vision, excellent accuracy, and resistance to fluctuations in light intensity, laser-based SLAM is extensively utilized in robotic applications. Specifically, 2D Lidar SLAM is more economical than 3D Lidar SLAM and is predominantly utilized for indoor robotics [11].

Two-dimensional Lidar SLAM is generally categorized into four components: a front-end odometer, back-end optimization, mapping, and loop detection, as seen in (Fig. 1). Initially, 2D Lidar acquires ambient data and transmits point cloud information; thereafter, the front-end odometer receives and processes this data, calculates the pose via matching, and produces local sub-maps. Subsequently, posture information and motion restrictions are employed in the back-end optimization module through techniques like as filtering and graph optimization. It eradicates accumulated faults in the local map, further enhances the robot's pose and sub-maps, and rectifies the global map. Ultimately, loop detection amalgamates the robot's pose with interframe data to ascertain if the robot has traversed a comparable location. It eradicates map distortion resulting from cumulative inaccuracies to attain a globally uniform map.

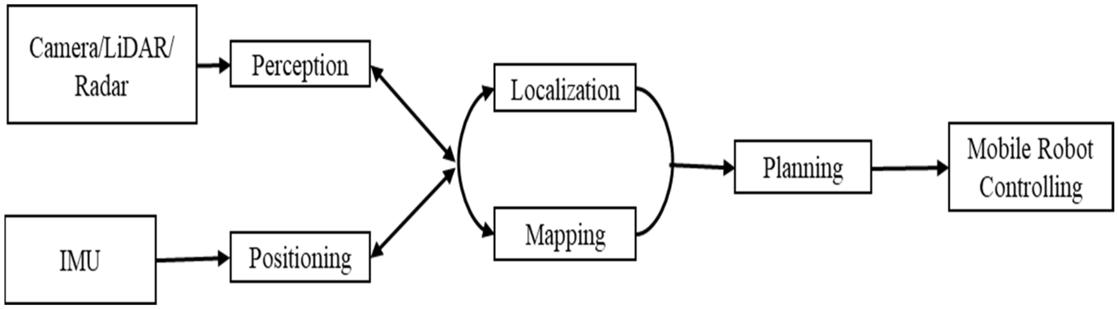


Fig.1 Autonomous mobile robot system architecture

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

SLAM algorithms use probabilistic methods parametric filters (e.g., Kalman filters), non-parametric filters (e.g., Particle filters), and optimization techniques to process Lidar data, enabling accurate 2D or 3D mapping depending on the system configuration.

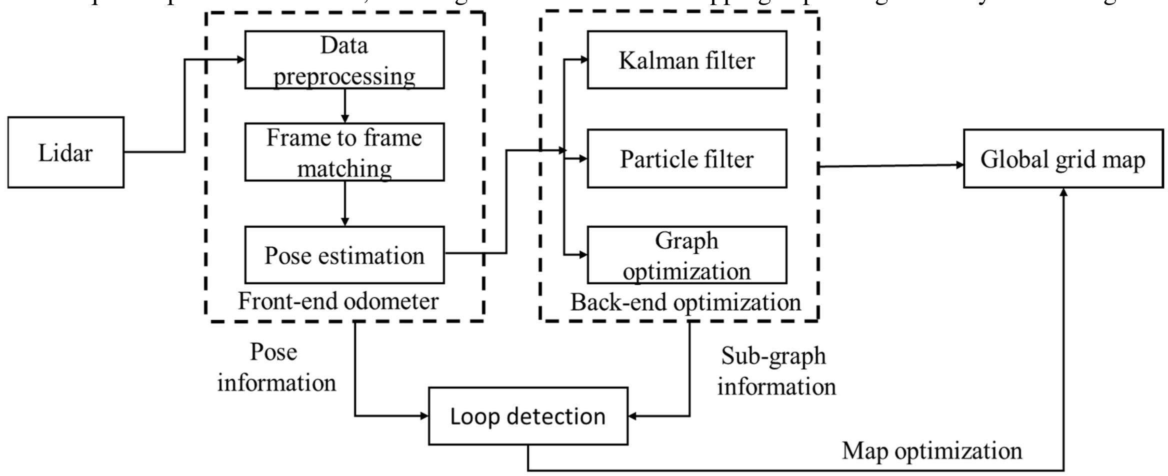


Fig.2 Key algorithms in Lidar-based localization and mapping

II. LITERATURE REVIEW

Simultaneous Localization and Mapping (SLAM) refers to the difficulty of creating a map for an unknown area using onboard sensors while also addressing the localization problem. Filter-based SLAM is one of the most common techniques to solving this challenge. This approach is based on Bayesian filtering theory. It comprises two major steps: (1) the prediction stage, in which robot localization and map state are updated using prior system state information and input control instructions; (2) the measurement update, in which current sensor data is compared to expected system state to produce new system state predictions. This strategy has several implementations. Earlier SLAM systems were based on the Extended Kalman filter (EKF-based SLAM,) or the Particle Filter (e.g., FastSLAM. Surveys provide an introduction to the basics of SLAM systems, while provides more specific information regarding filter-based systems.

The SLAM problem may be handled by using various sensors, and the selection of appropriate sensors is especially important in the effective operation of Unmanned Ground Vehicles (UGVs) due to the limited autonomous resources. Most mobile robots incorporate an Inertial Measurement Unit (IMU), which has accelerometers, gyroscopes, and magnetometers that measure orientation, angular velocity, and acceleration, respectively. This technique to mobile robot localization utilizing simply an Inertial Navigation System (INS) might result in considerable navigation mistakes. As a result, Lidar is typically used as the primary sensor for indoor robotic navigation and SLAM. 2D Lidar SLAM systems are currently available in a variety of packages, including GMapping (which uses Rao-Blackwellized particle filer to learn grid maps from 2D Lidar data), Hector SLAM (another popular ROS-based SLAM), and Cartographer (one of the most recent systems).

Monocular and stereo cameras, on the other hand, are ideal low-cost passive sensors that may efficiently address the SLAM issue by acting as a single source of information about an environment, a technique known as Visual SLAM (V-SLAM. All V-SLAM approaches fall into two categories: (1) feature-based, which employ specific features to construct maps, and (2) direct, which operate with complete pictures. Previous studies on visual navigation used a binocular stereo camera as well as a monocular camera (known as MonoSLAM. Over the past decade, numerous methods have been proposed, including Parallel Tracking and Mapping (PTAM), Regularized Monocular Depth Estimation (REMODE), Oriented FAST and Rotated BRIEF (ORB-SLAM), Dense Tracking and Mapping in Real-Time (DTAM), Large-Scale Direct Monocular SLAM (LSD-SLAM), Large-Scale Direct SLAM with Stereo Cameras (Stereo LSD-SLAM[12]). Common characteristics include low resilience to harsh circumstances, experimentation, poor performance in environments with limited geometric variation, and sensitivity to pure rotations [13,14]. These solutions do not give the metric information that some applications demand (also known as scale ambiguity). The answer to this challenge is being discussed in the computer vision and robotics communities. Thus, for example, the study [15] suggested employing additional sensors such as IMU, whereas authorsoffered a method for recovering metric information using knowledge on the geometric sensor layout. Other major factors that might contribute to SLAM quality are map initialization [16,17], loop closure [18] (which adjusts for mistakes accrued during UGV laps), and the kind of camera shutter employed [19].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

In this research, we primarily focus on a real-world robotics application for current SLAM systems that are implemented in Robot Operating System (ROS). For this purpose, we study the following SLAM systems: (a) 2D Lidar: Gmapping, Hector SLAM, Cartographer; (b) monocular camera: LSD SLAM, ORB SLAM, DSO; and (c) stereo camera: ZEDfu, RTAB map, ORB SLAM, S-PTAM. Table 1 provides a quick overview of the ROS-based SLAM algorithms investigated in this study.

Table 1 ROS-based simultaneous localization and mapping (SLAM) systems studied in this research

System	Sensor
Gmapping	2D Lidar
Parallel tracking and mapping (PTAM)	Mono
Hector SLAM	2D Lidar
Semi-direct visual odometry (SVO)	Mono
Large scale direct monocular SLAM (LSD SLAM)	Mono
Real-time appearance-based mapping (RTAB map)	Stereo
ORB SLAM	Mono, stereo
Dense piecewise parallel tracking and mapping (DPPTAM)	Mono
Direct sparse odometry (DSO)	Mono
Cartographer	2D Lidar
Stereo parallel tracking and mapping (S-PTAM)	Stereo

III. SYSTEM IMPLEMENTATION

The symbolic definitions for the robot are labeled in Fig. 3. A 2D world frame $\{W\}$ and a robot frame$\{R\}$, located at the center of mass, are established to describe the robot's position $\boldsymbol{S}$ and orientation θ_z at discrete timet_k. Key parameters include the linear velocities v_x and v_y of the robot with respect to frame {R}, the angular velocityw_z, the angular velocity $w_i$of the ith wheel, the wheel radius r, half the distance between the front wheels represented by L_x, and half the distance between the front and rear wheels designated as L_y. There are two sources of data to estimate the robot's pose: the wheel encoder. The speed of the wheels can be read directly from the encoder to compute the robot's velocity with respect to the robot frame {R}at timet_k, based on the kinematic model of the four-mecanum-wheeled robot using the following Eq. (1)

$$\begin{bmatrix} v_{k,x}^{R} \\ v_{k,y}^{R} \\ w_{k,z} \end{bmatrix} = \frac{r}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ \frac{-1}{L_x+L_y} & \frac{1}{(L_x+L_y)} & \frac{-1}{L_x+L_y} & \frac{1}{L_x+L_y} \end{bmatrix} \begin{bmatrix} w_{k,1} \\ w_{k,2} \\ w_{k,3} \\ w_{k,4} \end{bmatrix} \quad (1)$$

Subsequently, given the orientation at the z-axis θ_z the robot velocity with respect to the world frame {W} at discrete time t_k can be described by Eq. (2).

$$\begin{bmatrix} v_{k,x} \\ v_{k,y} \end{bmatrix} = \begin{bmatrix} \cos(\theta_{k,z}) & -\sin(\theta_{k,z}) \\ \sin(\theta_{k,z}) & \cos(\theta_{k,z}) \end{bmatrix} \begin{bmatrix} v_{k,x}^{R} \\ v_{k,y}^{R} \end{bmatrix} \quad (2)$$

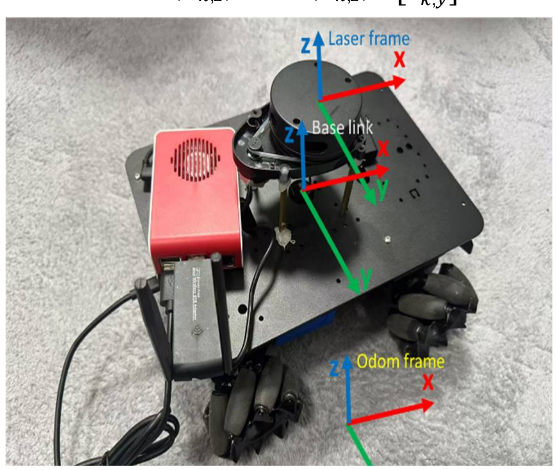


Fig.3 Symbolic pose definitions of robotic system

The robot's pose estimation requires proper calibration before processing. In an ideal IMU, the three-axis accelerometers are orthogonal to each other, forming a Cartesian coordinate system. Each accelerometer measures the acceleration along its respective axis, while the gyroscope measures the angular velocity around that axis.However, manufacturing process errors may cause the three axes to deviate from perfect orthogonally. As a result, the accelerometer and gyroscope coordinate systems may not completely overlap, and individual sensors may exhibit inaccuracies.

In this study, we follow the Ferraris Calibration procedure to correct sensor offsets and misalignments. After performing, the robot pose estimation data involves processing the corrected measurements. The acceleration and gyroscope data are used to calculate its orientation and displacement. The gyroscope provides the angular velocity ω, about each axis, which can be integrated over time to obtain the robot's orientation θ, relative to an initial orientation. This is represented by Eq. (3):

$$\theta_{k,z} = \theta_{k-1,z} + w_{k,z}\Delta t_k \qquad (3)$$

where $\boldsymbol{w_{k,z}}$is the reading along the z-axis at time k. For positional estimation, the acceleration measurements, which include both translational acceleration and the gravitational acceleration component, must first have the gravity vector a_g subtracted to isolate the translational acceleration. Afterward, the resulting acceleration can be integrated twice with respect to time to estimate the displacement s, as shown below:

$$\begin{bmatrix} v_{k,x} \\ v_{k,y} \end{bmatrix} = \begin{bmatrix} v_{k-1,x} \\ v_{k-1,y} \end{bmatrix} + \begin{bmatrix} a_{k,x} - a_{g,x} \\ a_{k,y} - a_{g,y} \end{bmatrix} \Delta t_k \qquad (4)$$

$$\begin{bmatrix} s_{k,x} \\ s_{k,y} \end{bmatrix} = \begin{bmatrix} s_{k-1,x} \\ s_{k-1,y} \end{bmatrix} + \begin{bmatrix} v_{k,x} \\ v_{k,y} \end{bmatrix} \Delta t_k + \frac{1}{2} \begin{bmatrix} a_{k,x} - a_{g,x} \\ a_{k,y} - a_{g,y} \end{bmatrix} \Delta t_k^2 \qquad (5)$$

However, due to the intrinsic noises and biases of sensors, solely relying on encoder or IMU data for pose estimation over longer periods introduces cumulative errors, known as drift. Therefore, estimation data is often fused with other sensor data using sensor fusion algorithms.

The Kalman Filter (KF) is used to achieve a more accurate and robust pose estimation over time. Given the state vector$x_k = \left[s_x v_x a_x s_y v_y a_y \theta_z w_z\right]^T$, the state transition and measurement functions can be derived to describe the system's dynamics and how these dynamics are measured, respectively.

The YDLIDAR X3 utilized in this study employs laser triangulation technology, which uses an infrared laser beam to capture environmental information. As the mobile robot moves through the working environment to gather data, it is assumed that $\boldsymbol{z_i}$ the position of each laser point, denoted as (ρ_i, α_i), is determined for each observation, where i=1, 2, 3… N and N is the total number of laser points measured in a single scan. The raw point cloud data obtained from the Lidarscan is preprocessed, and the data in polar coordinates is then converted to the obstacle's coordinates in the Lidar coordinate system using Equation

$$\begin{cases} x = \rho \cos \alpha \\ y = \rho \sin \alpha \end{cases} \qquad (6)$$

The Lidar communicates with the host computer via the serial port, transmitting the collected data for further processing, thereby completing the data acquisition. This process is repeated sequentially for each angle, with the Lidar collecting measurement data from every direction. The data from each direction is then correlated and fused to determine the distances between the detected obstacles and the mobile robot within the effective range. After transforming the coordinates into the Cartesian coordinate system and performing the imaging process, point cloud data within a specific range is generated. The host computer can then use this point cloud data to detect and match features of the current environment. The Lidar employs a low-power (<5mW) infrared laser as its light source, which is activated using modulated pulses. One safety feature that enhances convenience is its brief laser emission, ensuring it does not damage humans or pets.

Table2 Measurement performance of the YDLIDAR X3

Item	Unit	Min	Typical	Max
Distance range	Meter(m)	N/A	Range is 12 m	12
Angular range	Degree	N/A	0-360	N/A
Distance resolution	mm	N/A	< 0.5 < 1% of distance	N/A
Angular resolution	Degree	N/A	<1	N/A

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Sample duration	Milliseconds	N/A	0.125	N/A
Sample frequency	Hz	N/A	>8000	8010
Scan rate	Hz	N/A	5.5	10
Laser wavelength	Nanometre(nm)	775	785	795

Following the completion of the ROS integration, the subsequent phase for the application outlined in this work necessitates the construction of the environmental map. The integration of ROS with sensor data has significantly enhanced the convenience and analytical capability of the autonomous features of robots. ROS utilizes measurement data along with topic and frame modifications to assist the robot in accurately determining its position and orientation inside the map. To achieve this, ROS employs the Hector SLAM (simultaneous localization and mapping) technology to its advantage. In the subsequent subsection, we will examine the application of SLAM and the role of the Gmapping method, which also utilizes SLAMin constructing a map of any environment using laser scan data.

The total displacement of the robot, assuming it follows a small arc-like movement, is given by the average of both wheel displacements:

$$D = \frac{D_L + D_R}{2} \qquad (7)$$

The change in orientation (heading) of the robot is calculated as:

$$\Delta\theta = \frac{D_R - D_L}{L} \qquad (8)$$

This equation shows that when both wheels move the same distance, $\Delta\theta$ is zero, meaning the robot moves in a straight line. However, if one wheel moves more than the other, the robot turns in the direction of the slower-moving wheel.Using these values, the updated robot position in the 2D plane is calculated as follows:

$$x' = x + D\cos\left(\theta + \frac{\Delta\theta}{2}\right) \qquad (9)$$

$$y' = y + D\sin\left(\theta + \frac{\Delta\theta}{2}\right) \qquad (10)$$

where equations assume that robot moves in small increments, updating its position iteratively.

IV. EXPERIMENTAL WORK

Hector SLAM follows a similar principle to other SLAM algorithms, but it specifically employs a 2D SLAM approach. It integrates robust laser scans from the YDLIDAR X3 with a 3D navigation approach using an inertial sensing system, facilitated by the Extended Kalman Filter (EKF). The algorithm is designed to calculate the robot's 6 degrees of freedom (6DOF) pose during motion, leveraging the high update rate from the 2D Lidar scans. The key advantage of using Hector SLAM over traditional SLAM algorithms like Gmapping is that it doesn't rely on odometry data from wheel encoders to process and generate the map. This allows for greater flexibility, enabling users to simultaneously map the environment and localize the robot without being constrained by the physical limitations of the robot's wheels.

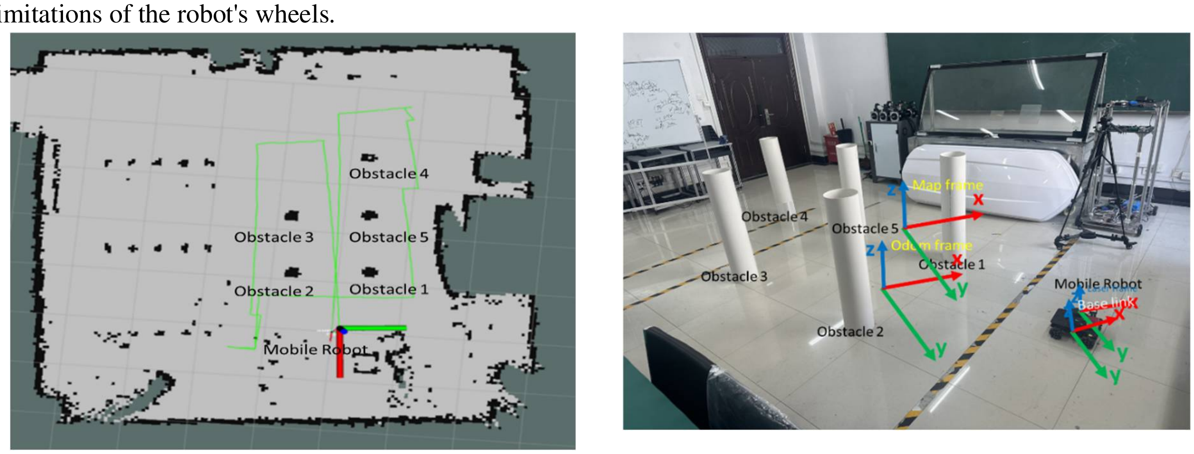


Fig.4 Map building using Hector SLAM in the school's lab

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Experiments are conducted in an environment of the school's lab (shown in Fig.4). To replicate robotic platform movement, the mobile robot with a Lidar sensor and Raspberry Pi is manually maneuvered to survey the surrounding region. The Lidar data is taken from the Raspberry Pi and transferred to a remote computer for processing and visualization using the rviz program.

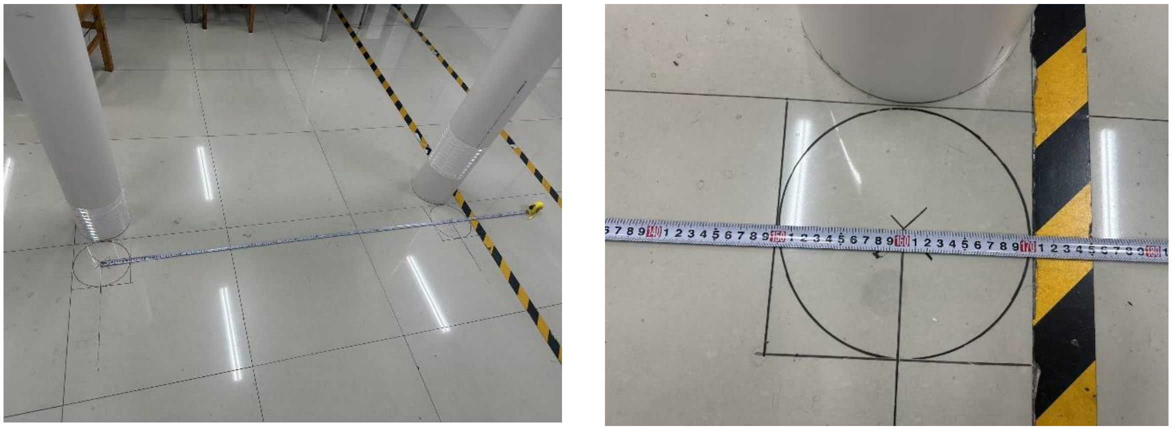


Fig.5 Physical measurement of the distance between two landmarks in the lab

Table 3 Comparison of distance measurements

Test	Normal value (m)	Measured value (m)	Error (%)	RMSE (m)	MAPE (%)	MAE (m)
1	1.60	1.54	3.75	0.06	3.75	0.06
2	1.60	1.45	9.37	0.15	9.375	0.15
3	1.60	1.54	3.75	0.06	3.75	0.15
4	1.60	1.53	4.37	0.07	4.375	0.07
5	1.60	1.57	1.88	0.03	1.875	0.03
6	1.60	1.57	1.88	0.03	1.875	0.03
7	1.60	1.58	1.25	0.02	1.25	0.02
8	1.60	1.59	0.62	0.01	0.625	0.01
9	1.60	1.54	3.75	0.06	3.75	0.06
10	1.60	1.57	1.88	0.03	1.875	0.03
		Average Error	**3.26**			

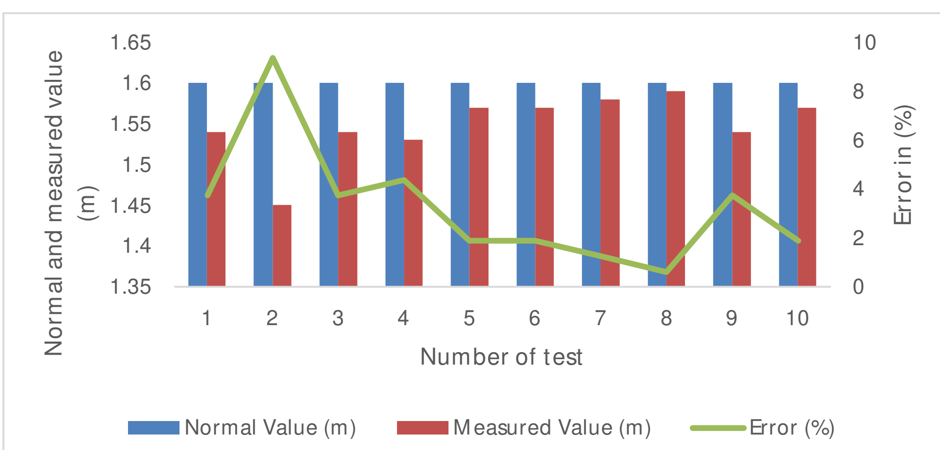


Fig.6 Error metrics

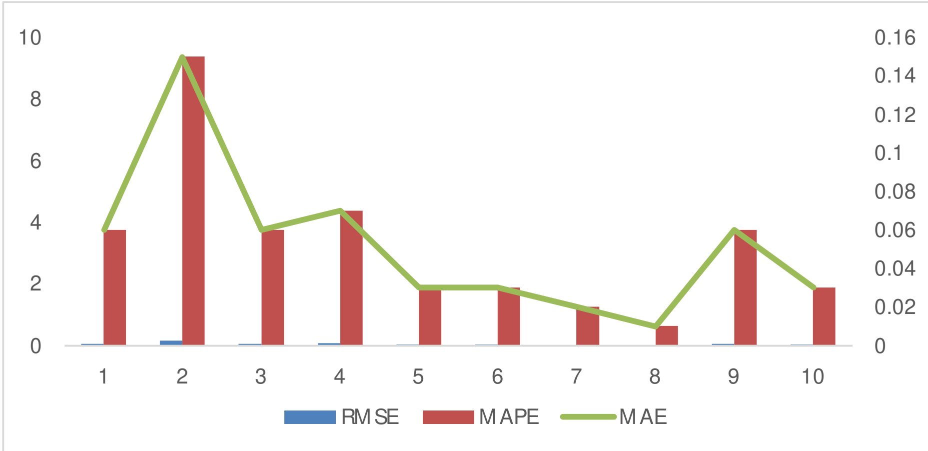


Fig.7 Measurement comparison

The data compares the standard values and the measured distances for 10 tests, including the associated percentage errors that shown in Fig.5. Analysis of the computed performance metrics RMSE, MAE, and MAPE reveals that the errors across the tests are predominantly minimal, suggesting that the measurements are quite precise in table 3 and Fig.6. The Root Mean Square Error (RMSE) values of Fig.7approach 0, indicating a strong correlation between the anticipated and actual values, with little discrepancies. RMSE is especially responsive to significant mistakes, hence a low number indicates the general accuracy of the measurements. The Mean Absolute Error (MAE), which calculates the average of the absolute discrepancies between the actual and measured values, is likewise notably low, further validating that the measured distances are near the true values, with negligible error.

The Mean Absolute Percentage Error (MAPE) of the result indicates an average error of approximately 3.25%, reflecting a comparatively low percentage and proving that the model or measurement system consistently yields data with a high level of accuracy throughout the testing. The findings demonstrate that the measurement system is dependable, exhibiting only minor discrepancies, and is suitable for applications necessitating accurate distance measurements.

V. CONCLUSIONS

In conclusion, combining YDLIDAR X3 with ROS-based mapping makes a strong base for low-cost, accurate, and efficient robotic mapping and localization systems that could be used in the future and make a big difference.

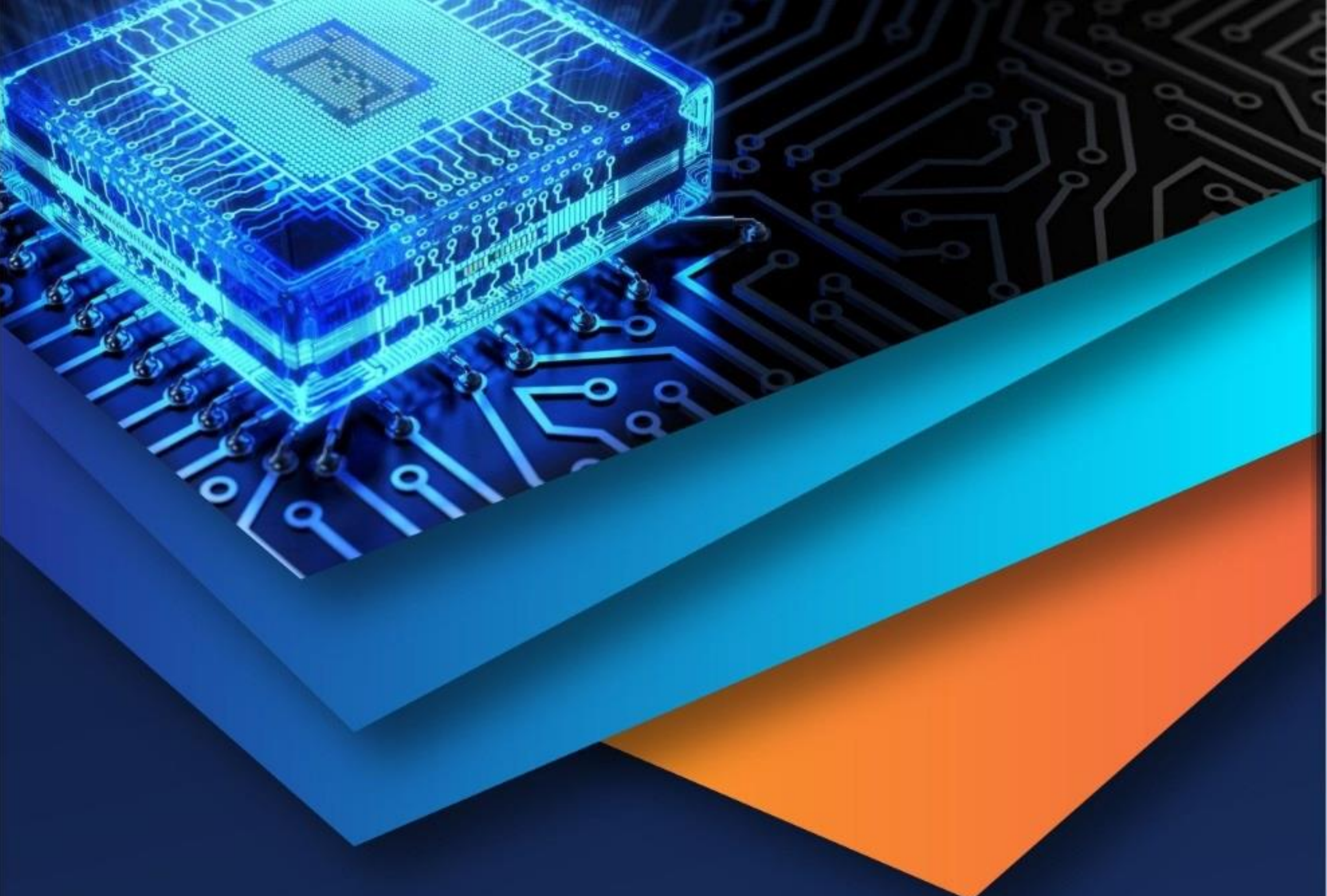
VI. ACKNOWLEDGMENTS

The authors express their gratitude torobotic lab mates at Zhejiang University of Science and Technology, for their collaboration, technical support, and encouragement, which significantly enhanced the success of this project.

REFERENCES

[1] Liu, B.; Xiao, X.; Stone, P. A Lifelong Learning Approach to Mobile Robot Navigation. IEEE Robot. Autom. Lett. 2021, 6, 1090–1096.
[2] De Heuvel, J.; Zeng, X.; Shi, W.; Sethuraman, T.; Bennewitz, M. Spatiotemporal Attention Enhances Lidar-Based Robot Navigation in Dynamic Environments. IEEE Robot. Autom. Lett. 2024, 9, 4202–4209.
[3] Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. IEEE Trans. Robot. 2016, 32, 1309–1332.
[4] Xu, X.; Zhang, L.; Yang, J.; Cao, C.; Wang, W.; Ran, Y.; Tan, Z.; Luo, M. A Review of Multi-Sensor Fusion SLAM Systems Based on 3D LIDAR. Remote Sens. 2022, 14, 2835.
[5] Zhang, J.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-time. Robot. Sci. Syst. 2014, 2, 9.
[6] Shan, T.; Englot, B. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765.
[7] Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Rus, D. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24–30 October 2020; pp. 5135–5142.
[8] Davison, A.J.; Reid, I.D.; Molton, N.D.; Stasse, O. MonoSLAM: Real-Time Single Camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. 2007, 29, 1052–1067.
[9] Mur-Artal, R.; Tardos, J.D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. IEEE Trans. Robot. 2017, 33, 1255–1262.

[10] Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. IEEE Trans. Robot. 2018, 34, 1004–1020.
[11] Jiang, G.; Yin, L.; Jin, S.; Tian, C.; Ma, X.; Ou, Y. A Simultaneous Localization and Mapping (SLAM) Framework for 2.5D Map Building Based on Low-Cost Lidar and Vision Fusion. Appl. Sci. 2019, 9, 2105.
[12] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct slam with stereo cameras," in Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on. IEEE, 2015, pp. 1935–1942.
[13] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," Proceedings - IEEE International Conference on Robotics and Automation, pp. 15–22, 2014.
[14] M. Labbé and F. Michaud, "Online global loop closure detection for large-scale multi-session graph-based SLAM," IEEE Int. Conference on Intelligent Robots and Systems, pp. 2661–2666, 2014.
[15] A. Concha and J. Civera, "DPPTAM: Dense piecewise planar tracking and mapping from a monocular sequence," IEEE International Conference on Intelligent Robots and Systems, pp. 5686–5693, 2015.
[16] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 3, pp. 611–625, 2018.
[17] T. Whelan, S. Leutenegger, R. Salas Moreno, B. Glocker, and A. Davison, "ElasticFusion: Dense SLAM without A Pose Graph," Robotics: Science and Systems XI, 2015.
[18] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: Real-time dense monocular slam with learned depth prediction," in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 2, 2017.
[19] T. Pire, T. Fischer, G. Castro, P. DeCristóforis, J. Civera, and J. JacoboBerlles, "S-PTAM: Stereo Parallel Tracking and Mapping," Robotics and Autonomous Systems, vol. 93, pp. 27–42, 2017.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)